EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	2	specification near3 translat$4 same integrated adj circuit same element	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/07 11:58
S1	3	(sign and symptom) same user near3 interface same agent	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/07 11:58
S2	2	"20010056435"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/08/31 13:37
S3	0	inertial adj rejection same simulat$3	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/08/31 13:38
S4	3	inertial adj rejection	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/08/31 13:41
S5	12	atomic adj transaction same circuit	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/05 16:40
S6	0	atomic adj transaction same object same null	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/08/31 13:42
S7	86	atomic adj transaction same object	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/08/31 13:43
S8	71	S7 and @ad<"20030314"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/08/31 13:43

EAST Search History

S9	4	(sign and symptom) same user near3 interface and biological and chemical	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/08/31 14:38
S10	9	((sign and symptom) same user near3 interface and (biological or chemical))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/08/31 14:36
S11	1	(sign and symptom) same GUI near3 interface and biological and chemical	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/08/31 14:39
S12	22	(sign and symptom) same interface and biological and chemical	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/08/31 15:58
S13	6	(sign and symptom) near6 input$3 same hazard$3 and biological and chemical	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/08/31 17:04
S14	3	2004-178372	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/08/31 17:05
S15	86	configuration adj command same state adj machine	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/05 16:45
S16	43	configuration near6 design same state adj machine	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/05 17:32
S17	4	2004/0006584	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/05 17:32
S18	2	"20040006584"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/05 17:33

EAST Search History

S19	1	S18 and packet	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/09/05 17:33

Terms used **integrated circuit specification translation** Found **66,263** of **185,030**

Sort results by: relevance
Display results: expanded form
Save results to a Binder
Search Tips
Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next
Best 200 shown Relevance scale

1 Special session: hierarchical design and design space exploration of analog integrated circuits: Deterministic approaches to analog performance space exploration (PSE)
Daniel Mueller, Guido Stehr, Helmut Graeb, Ulf Schlichtmann
June 2005 **Proceedings of the 42nd annual conference on Design automation**
Publisher: ACM Press
Full text available: pdf(256.58 KB) Additional Information: full citation, abstract, references, index terms

Performance space exploration (PSE) determines the range of feasible performance values of a circuit block for a given topology and technology. In this paper, we present two deterministic approaches for PSE. One approximates the feasible performance space based on linearized circuit models and is suitable for investigating a large number of performances. The other one computes discretizations of the Pareto front of competing performances. In addition, a motivation and application of PSE using a ...

Keywords: analog integrated circuits, fourier motzkin elimination, pareto optimization, performance space exploration

2 A VHDL-AMS compiler and architecture generator for behavioral synthesis of analog systems
Alex Doboli, Ranga Vemuri
January 1999 **Proceedings of the conference on Design, automation and test in Europe**
Publisher: ACM Press
Full text available: pdf(104.79 KB) Additional Information: full citation, citings, index terms

3 Efficient Generation of Monitor Circuits for GSTE Assertion Graphs
Alan J. Hu, Jeremy Casas, Jin Yang
November 2003 **Proceedings of the 2003 IEEE/ACM international conference on Computer-aided design**
Publisher: IEEE Computer Society
Full text available: pdf(159.73 KB) Additional Information: full citation, abstract, index terms

Generalized symbolic trajectory evaluation (GSTE) is a powerful,new method for formal verification that combines the industrially-provenscalability and capacity of classical

symbolic trajectoryevaluation with the expressive power of temporal-logic modelchecking. GSTE was originally developed at Intel and hasbeen used successfully on Intel's next-generation microprocessors.However, the supporting algorithms and tools for GSTE are stillrelatively immature.GSTE specifications are given as assertion ...

4 Delay-insensitive interface specification and synthesis

Mark B. Josephs, Dennis Furey

January 2000 **Proceedings of the conference on Design, automation and test in Europe**

Publisher: ACM Press

Full text available: pdf(49.75 KB) Publisher Site

Additional Information: full citation, references, index terms

5 SHILPA: a high-level synthesis system for self-timed circuits

Venkatesh Akella, Ganesh Gopalakrishnan

November 1992 **Proceedings of the 1992 IEEE/ACM international conference on Computer-aided design**

Publisher: IEEE Computer Society Press

Full text available: pdf(511.65 KB) Additional Information: full citation, references, citings, index terms

6 The application accelerator illustration system

Michael S. Miller, Howard Cunningham, Chan Lee, Steven R. Vegdahl

June 1986 **ACM SIGPLAN Notices , Conference proceedings on Object-oriented programming systems, languages and applications OOPLSA '86**, Volume 21 Issue 11

Publisher: ACM Press

Full text available: pdf(629.30 KB) Additional Information: full citation, abstract, references, citings, index terms

The Application Accelerator Illustration System is a prototype of an integrated CAD environment that supports the development of application-specific integrated circuits. The current implementation features a hardware description language compiler, timing analyzer, functional simulator, waveform tracer, and data path place and route facility. The system is implemented in Smalltalk-80™.

7 Computer integration: a co-requirement for effective inter-organization computer network implementation

Paul Hart, D. Estrin

September 1990 **Proceedings of the 1990 ACM conference on Computer-supported cooperative work**

Publisher: ACM Press

Full text available: pdf(1.06 MB) Additional Information: full citation, abstract, references, citings, index terms

Inter-organization computer networks (IONS) provide significant opportunities for improving coordination between firms engaged in mutually dependent activities. This research paper focuses on how IONS affect information processing requirements, and production and transaction costs when they interconnect firms with internally integrated computer systems and when they are used only as substitutes for conventional media. We conclude that significant improvements in inter- organization coordina ...

8 A prototype framework for knowledge-based analog circuit synthesis

R. Harjani, R. A. Rutenbar, L. R. Carley

June 1988 **Papers on Twenty-five years of electronic design automation**
Publisher: ACM Press
Full text available: pdf(1.06 MB) Additional Information: full citation, references, index terms

9 Behavioral synthesis of field programmable analog array circuits
Haibo Wang, Sarma B. K. Vrudhula
October 2002 **ACM Transactions on Design Automation of Electronic Systems (TODAES)**, Volume 7 Issue 4
Publisher: ACM Press
Full text available: pdf(519.64 KB) Additional Information: full citation, abstract, references, index terms

This article presents methods to translate a behavioral-level analog description into a Field Programmable Analog Array (FPAA) implementation. The methods consist of several steps that are referred to as function decomposition, macrocell synthesis, placement and routing, and postplacement simulation. The focus of this article is on the first three steps. The function decomposition step deals with decomposing a high-order system function into a set of lower-order functions. We present an efficien ...

Keywords: Programmable circuits, analog synthesis

10 An interactive design automation system
Stephen Y. H. Su
June 1973 **Proceedings of the 10th workshop on Design automation**
Publisher: IEEE Press
Full text available: pdf(851.96 KB) Additional Information: full citation, abstract, references, citings, index terms

An interactive design automation system is presented which, after complete implementation, will allow the designer to check the determinacy and dead locks of the system before implementation. The design can be evaluated at various levels and modified interactively. The designer enters his design specification using either graphical representation or design language statements. The translator accepts the input and produces a data base for both the simulator and the logic synthesizer. The syn ...

11 A prototype framework for knowledge-based analog circuit synthesis
R. Harjani, R. A. Rutenbar, L. R. Carley
October 1987 **Proceedings of the 24th ACM/IEEE conference on Design automation**
Publisher: ACM Press
Full text available: pdf(1.07 MB) Additional Information: full citation, abstract, references, citings, index terms

An organization for a knowledge-based analog circuit synthesis tool is described. Analog circuit topologies are represented as a hierarchy of functional blocks; a planning mechanism is introduced to translate performance specifications between levels in this circuit hierarchy. A prototype implementation, OASYS, synthesizes sized transistor schematics for simple CMOS operational amplifiers from performance specifications and process parameters, and demonstrates the workability of the approac ...

12 CANLOGS: a logic gate simulator for all seasons
R. Mark Meyer
March 1996 **ACM SIGCSE Bulletin , Proceedings of the twenty-seventh SIGCSE technical symposium on Computer science education SIGCSE '96**, Volume 28 Issue 1
Publisher: ACM Press

Full text available: pdf(551.75 KB) Additional Information: full citation, references, citings, index terms

13 A survey on bit dimension optimization strategies of microprograms
Sunil R. Das, Amiya R. Nayak
November 1990 **Proceedings of the 23rd annual workshop and symposium on Microprogramming and microarchitecture**
Publisher: IEEE Computer Society Press
Full text available: pdf(1.33 MB) Additional Information: full citation, abstract, references

Microprogram optimization is one way to increase efficiency, and optimization can be crucial in some applications. Optimization refers to a reduction of execution time of microprograms, or of the control store size, B × W, where W represents the word dimension of the control store which is the number of words of control store required for certain application, and B represents the bit dimension which is the number of bits per word of control store. The various optimization strategies c ...

14 A technique for synthesizing distributed burst-mode circuits
Prabhakar Kudva, Ganesh Gopalakrishnan, Hans Jacobson
June 1996 **Proceedings of the 33rd annual conference on Design automation**
Publisher: ACM Press
Full text available: pdf(108.03 KB) Additional Information: full citation, references, citings, index terms

15 The rectangle placement language
John Alan Roach
June 1984 **Proceedings of the 21st conference on Design automation**
Publisher: IEEE Press
Full text available: pdf(539.63 KB) Additional Information: full citation, abstract, references, citings, index terms

Constraint-based symbolic layout of VLSI designs has received growing attention during the past few years. Several systems have been developed which can offer graphical or textual media for the expression of designs and can provide automatic compaction to technology specific design rules. The ability of these systems to allow for the expression of more generalized relationships is a topic open for further research. RPL is a constraint-based symbolic layout language intended to ...

16 ADL: An algorithmic design language for integrated circuit synthesis
W. H. Evans, J. C. Ballegeer, Nguyen H. Duyet
June 1984 **Proceedings of the 21st conference on Design automation**
Publisher: IEEE Press
Full text available: pdf(659.93 KB) Additional Information: full citation, abstract, references, citings, index terms

The Algorithmic Design Language (ADL), provides a means to procedurally describe the functional, circuit, schematic and mask aspects of integrated circuits. The constructs of this language have been coded in the C language and are intended for application to IC design. C programs that incorporate ADL routines are executed to build a data base from which CIF files, input files to circuit simulation programs or a textual representation of ADL's own highly structured data base can be generated ...

17 Advances in synthesis: Resynthesis and peephole transformations for the optimization of large-scale asynchronous systems

Tiberiu Chelcea, Steven M. Nowick
June 2002 **Proceedings of the 39th conference on Design automation**
Publisher: ACM Press
Full text available: pdf(112.73 KB) Additional Information: full citation, abstract, references, citings

Several approaches have been proposed for the syntax-directed compilation of asynchronous circuits from high-level specification languages, such as Balsa and Tangram. Both compilers have been successfully used in large real-world applications; however, in practice, these methods suffer from significant performance overheads due to their reliance on straightforward syntax-directed translation.This paper introduces a powerful new set of transformations, and an extended channel-based language to su ...

18 Mixing Global and Local Competition in Genetic Optimization based Design Space Exploration of Analog Circuits

Abhishek Somani, P. P. Chakrabarti, Amit Patra
March 2005 **Proceedings of the conference on Design, Automation and Test in Europe - Volume 2 DATE '05**
Publisher: IEEE Computer Society
Full text available: pdf(202.29 KB) Additional Information: full citation, abstract, citings, index terms

The knowledge of optimal design space boundaries of component circuits can be extremely useful in making good subsystem-level design decisions which are aware of the parasitics and other second-order circuit-level details. However, direct application of popular Multi-objective genetic optimization algorithms were found to produce Pareto fronts with poor diversity for analog circuits problems. This work proposes a novel approach to control the diversity of solutions by paritioning the solution sp ...

19 A Burst-Mode Oriented Back-End for the Balsa Synthesis System

T. Chelcea, S. Nowick, A. Bardsley, D. Edwards
March 2002 **Proceedings of the conference on Design, automation and test in Europe**
Publisher: IEEE Computer Society
Full text available: pdf(203.66 KB) Publisher Site Additional Information: full citation, abstract, citings

This paper introduces several new component clustering techniquesfor the optimization of asynchronous systems. In particular, novel"Burst-Mode aware" restrictions are imposed to limit the cluster sizesand to ensure synthesizability. A new control specification language,CH, is also introduced which facilitates the manipulation and optimizationof handshake control components. The new method has beenfully integrated into a comprehensive asynchronous synthesis package,Balsa. Experimental results on ...

20 ATPG-based logic synthesis: an overview

Chih-Wei Jim Chang, Malgorzata Marek-Sadowska
November 2002 **Proceedings of the 2002 IEEE/ACM international conference on Computer-aided design**
Publisher: ACM Press
Full text available: pdf(98.14 KB) Additional Information: full citation, abstract, references, index terms

The ultimate goal of logic synthesis is to explore implementation flexibility toward meeting design targets, such as area, power, and delay. Traditionally, such flexibility is expressed using "don't cares" and we seek the best implementation that does not violate them. However, the calculation and storing of don't care information is CPU and memory-intensive. In this paper, we give an overview of logic synthesis approaches based on techniques developed for Automatic Test Pattern Generation (ATPG ...

Terms used **programmable processing engines** Found **76,122** of **185,030**

Sort results by: relevance
Display results: expanded form
Save results to a Binder
Search Tips
Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next
Best 200 shown Relevance scale

1 The STARS process engine: language and architecture to support process capture and multi-user execution
Scott Arthur Moody
November 1994 **Proceedings of the conference on TRI-Ada '94**
Publisher: ACM Press
Full text available: pdf(1.43 MB) Additional Information: full citation, abstract, references, citings, index terms

Process-centered computing environments are currently in their infancy, with notable exceptions [1][19][21][22]. Two important components of envisioned environments are the language used to describe the processes, and the architecture for the language implementation and execution. These must support the multi-user emphasis of team work, process monitoring, process improvement, and automated execution.This paper reports on the STARS Process Engin ...

2 Concepts and implementation of a rule-based process engine
Burkhard Peuschel, Wilhelm Schäfer
June 1992 **Proceedings of the 14th international conference on Software engineering**
Publisher: ACM Press
Full text available: pdf(1.77 MB) Additional Information: full citation, references, citings, index terms

3 Engines build process abstractions
Christopher T. Haynes, Daniel P. Friedman
August 1984 **Proceedings of the 1984 ACM Symposium on LISP and functional programming**
Publisher: ACM Press
Full text available: pdf(497.54 KB) Additional Information: full citation, abstract, references, citings, index terms

Engines are a new programming language abstraction for timed preemption. In conjunction with first class continuations, engines allow the language to be extended with a time-sharing implementation of process abstraction facilities. To illustrate engine programming techniques, we implement a round-robin process scheduler. The importance of simple but powerful primitives such as engines is discussed.

4 Engineering e-learning systems (ELS): Carrying on the e-learning process with a

workflow management engine
Mirko Cesarini, Mattia Monga, Roberto Tedesco
March 2004 **Proceedings of the 2004 ACM symposium on Applied computing**
Publisher: ACM Press
Full text available: pdf(209.38 KB) Additional Information: full citation, abstract, references, index terms

In recent years e-learning systems have promised to change the way people learn. However open issues still remain, in particular actual e-learning environments do not consider learning activities as part of the process of learning. Thus, it is not possible to define structured courses and specify precise learning paths apt to guide learners through learning materials. In our approach, we define courses as workflows. By so doing we can exploit powerful procedural rules in order to define precise ...

Keywords: Workflow Mgmt Systems, e-learning, learning objects

5 A user-programmable vertex engine
Erik Lindholm, Mark J. Kligard, Henry Moreton
August 2001 **Proceedings of the 28th annual conference on Computer graphics and interactive techniques**
Publisher: ACM Press
Full text available: pdf(12.03 MB) Additional Information: full citation, abstract, references, citings, index terms

In this paper we describe the design, programming interface, and implementation of a very efficient user-programmable vertex engine. The vertex engine of NVIDIA's GeForce3 GPU evolved from a highly tuned fixed-function pipeline requiring considerable knowledge to program. Programs operate only on a stream of independent vertices traversing the pipe. Embedded in the broader fixed function pipeline, our approach preserves parallelism sacrificed by previous approaches. The programmer is presente ...

Keywords: graphics hardware, graphics systems

6 Research sessions: non-standard query processing: Fast computation of database operations using graphics processors
Naga K. Govindaraju, Brandon Lloyd, Wei Wang, Ming Lin, Dinesh Manocha
June 2004 **Proceedings of the 2004 ACM SIGMOD international conference on Management of data**
Publisher: ACM Press
Full text available: pdf(386.13 KB) Additional Information: full citation, abstract, references

We present new algorithms for performing fast computation of several common database operations on commodity graphics processors. Specifically, we consider operations such as conjunctive selections, aggregations, and semi-linear queries, which are essential computational components of typical database, data warehousing, and data mining applications. While graphics processing units (GPUs) have been designed for fast display of geometric primitives, we utilize the inherent pipelining and paralleli ...

Keywords: aggregation, graphics processor, query optimization, selection query, selectivity analysis, semi-linear query

7 Disconnected processes, mechanisms and architecture for mobile e-business
J. Sairamesh, S. Goh, I. Stanoi, S. Padmanabhan, C. S. Li
December 2004 **Mobile Networks and Applications**, Volume 9 Issue 6
Publisher: Kluwer Academic Publishers

Full text available: pdf(625.38 KB) Additional Information: full citation, abstract, references, index terms

With the tremendous advances in hand-held computing and communication capabilities, rapid proliferation of mobile devices, and decreasing device costs, we are seeing a growth in mobile e-business in various consumer and business markets. In this paper, we present a novel architecture and framework for end-to-end mobile e-business applications (e.g., point of sales). The architecture takes into consideration disconnection, application context, synchronization, transactions and failure recovery ...

Keywords: failure recovery, mobile commerce, mobile disconnection, mobile e-business, remote disconnection, seamless business transaction

8 Session 31: secure systems: Software architecture exploration for high-performance security processing on a multiprocessor mobile SoC

Divya Arora, Anand Raghunathan, Srivaths Ravi, Murugan Sankaradass, Niraj K. Jha, Srimat T. Chakradhar

July 2006 **Proceedings of the 43rd annual conference on Design automation DAC '06**

Publisher: ACM Press

Full text available: pdf(680.39 KB) Additional Information: full citation, abstract, references, index terms

We present a systematic methodology for exploring the security processing software architecture for a commercial heterogeneous multiprocessor system-on-chip (SoC) for mobile devices. The SoC contains multiple host processors executing applications and a dedicated programmable security processing engine. We developed an exploration methodology to map the code and data of security software libraries onto the platform, with the objective of maximizing the overall application-visible performance. Th ...

Keywords: computation offloading, software partitioning

9 GPGPU: general purpose computation on graphics hardware

David Luebke, Mark Harris, Jens Krüger, Tim Purcell, Naga Govindaraju, Ian Buck, Cliff Woolley, Aaron Lefohn

August 2004 **Proceedings of the conference on SIGGRAPH 2004 course notes SIGGRAPH '04**

Publisher: ACM Press

Full text available: pdf(63.03 MB) Additional Information: full citation, abstract

The graphics processor (GPU) on today's commodity video cards has evolved into an extremely powerful and flexible processor. The latest graphics architectures provide tremendous memory bandwidth and computational horsepower, with fully programmable vertex and pixel processing units that support vector operations up to full IEEE floating point precision. High level languages have emerged for graphics hardware, making this computational power accessible. Architecturally, GPUs are highly parallel s ...

10 Demos: ATCT: a Java framework that offers new approach to developing asynchronous processes

Serguei Mourachov

October 2003 **Companion of the 18th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications**

Publisher: ACM Press

Full text available: pdf(116.06 KB) Additional Information: full citation, abstract, index terms

The development of modern loosely coupled distributed applications requires extensive use of asynchronous processes. The ability to manipulate execution context could simplify development of such applications, helping to separate business logic from handling

asynchrony.This paper describes a framework that implements Execution Context Reification for Java Virtual Machine (JVM). The framework uses built-in secondary bytecode interpreter that provides access to Execution Context as a first class s ...

Keywords: Java framework, asynchronous processes, execution context reification

11 Real-time shading

Marc Olano, Kurt Akeley, John C. Hart, Wolfgang Heidrich, Michael McCool, Jason L. Mitchell, Randi Rost

August 2004 **Proceedings of the conference on SIGGRAPH 2004 course notes SIGGRAPH '04**

Publisher: ACM Press

Full text available: pdf(7.39 MB) Additional Information: full citation, abstract

Real-time procedural shading was once seen as a distant dream. When the first version of this course was offered four years ago, real-time shading was possible, but only with one-of-a-kind hardware or by combining the effects of tens to hundreds of rendering passes. Today, almost every new computer comes with graphics hardware capable of interactively executing shaders of thousands to tens of thousands of instructions. This course has been redesigned to address today's real-time shading capabili ...

12 Balancing performance and flexibility with hardware support for network architectures

Ilija Hadžić, Jonathan M. Smith

November 2003 **ACM Transactions on Computer Systems (TOCS)**, Volume 21 Issue 4

Publisher: ACM Press

Full text available: pdf(719.03 KB) Additional Information: full citation, abstract, references, index terms

The goals of performance and flexibility are often at odds in the design of network systems. The tension is common enough to justify an architectural solution, rather than a set of context-specific solutions. The Programmable Protocol Processing Pipeline (P4) design uses programmable hardware to selectively accelerate protocol processing functions. A set of field-programmable gate arrays (FPGAs) and an associated library of network processing modules implemented in hardware are augmented with so ...

Keywords: FPGA, P4, computer networking, flexibility, hardware, performance, programmable logic devices, programmable networks, protocol processing

13 Technology retargeting for IC layout

John Lakos

June 1997 **Proceedings of the 34th annual conference on Design automation DAC '97**

Publisher: ACM Press

Full text available: pdf(87.81 KB) Publisher Site Additional Information: full citation, abstract, references, index terms

The ability to recognize polygon-based layout as a collection ofobjects representing circuit elements connected by path-based wires,enables existing designs implemented using an older fabricationprocess to be reimplanted quickly in a new process.The approachtaken here, based on layout generator technology, is to create acollection of parameterized circuit objects that, with appropriatearguments, are able to represent the devices (e.g., transistors,contacts) implicitly described in the flattened ...

14 Games: Learning through game *modding*

Magy Seif El-Nasr, Brian K. Smith

January 2006 **Computers in Entertainment (CIE)**, Volume 4 Issue 1

Publisher: ACM Press
Full text available: pdf(694.54 KB) Additional Information: full citation, abstract, references, index terms

There has been a recent increase in the number of game environments or engines that allow users to customize their gaming experiences by building and expanding game behavior. This article describes the use of modifying, or *modding*, existing games as a means to learn computer science, mathematics, physics, and aesthetic principles. We describe two exploratory case studies of game modding in classroom settings to illustrate skills learned by students as a result of modding existing games. W ...

Keywords: game engines and classrooms, games and education, learning and design

15 Architectures: A programmable vertex shader with fixed-point SIMD datapath for low power wireless applications
Ju-Ho Sohn, Ramchan Woo, Hoi-Jun Yoo
August 2004 **Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware**
Publisher: ACM Press
Full text available: pdf(427.49 KB) Additional Information: full citation, abstract, references, index terms

The real time 3D graphics becomes one of the attractive applications for 3G wireless terminals although their battery lifetime and memory bandwidth limit the system resources for graphics processing. Instead of using the dedicated hardware engine with complex functions, we propose an efficient hardware architecture of low power vertex shader with programmability. Our architecture includes the following three features: *1*) a fixed-point SIMD datapath to exploit parallelism in vertex process ...

16 Commerce and Businesses: Self-managing, disconnected processes and mechanisms for mobile e-business
J. Sairamesh, S. Goh, I. Stanoi, C. S. Li, S. Padmanabhan
September 2002 **Proceedings of the 2nd international workshop on Mobile commerce**
Publisher: ACM Press
Full text available: pdf(351.90 KB) Additional Information: full citation, abstract, references, index terms

With the tremendous advances in hand-held computing and communication capabilities, rapid proliferation of mobile devices, and decreasing device costs, we are seeing a growth in mobile e-business in various consumer and business markets. In this paper, we present a novel architecture and framework for end-to-end mobile e-business applications such as purchasing, retail point of sales, and order management. The design takes into consideration disconnection, application context and failure modes t ...

Keywords: context-driven computing, disconnected computing, mobile computing, mobile e-business, self-managing systems, workflow

17 System design records: 40Gbps de-layered silicon protocol engine for TCP record
H. Shrikumar
March 2006 **Proceedings of the conference on Design, automation and test in Europe: Proceedings DATE '06**
Publisher: European Design and Automation Association
Full text available: pdf(293.43 KB) Additional Information: full citation, abstract, references

We present a de-layered protocol engine for termination of 40Gbps TCP connections using a reconfigurable FPGA silicon platform. This protocol engine is designed for a planned attempt at the Internet Speed Record. In laboratory demonstrations at 40Gbps, this core beat the previous record of 7.2Gbps by a factor of five. We present an aggressive cross-

layer optimization methodology and corresponding design-flow and tools used to implement this record-breaking TCP Protocol Engine.The 40Gbps TCP Offl ...

18 Memory optimization: Automatic data partitioning for the agere payload plus network processor

Steve Carr, Philip Sweany
September 2004 **Proceedings of the 2004 international conference on Compilers, architecture, and synthesis for embedded systems**
Publisher: ACM Press
Full text available: pdf(195.89 KB) Additional Information: full citation, abstract, references, index terms

With the ever-increasing pervasiveness of the Internet and its stringent performance requirements, network system designers have begun utilizing specialized chips to increase the performance of network functions. To increase performance, many more advanced functions, such as traffic shaping and policing, are being implemented at the network interface layer to reduce delays that occur when these functions are handled by a general-purpose CPU. While some designs use ASICs to handle network functio ...

Keywords: network processors, partitioning, scheduling

19 Turbine engine maintenance manpower and facility model

Gary E. Sundquist, Robert B. Whitegiver
January 1971 **Proceedings of the 5th conference on Winter simulation**
Publisher: ACM Press
Full text available: pdf(490.60 KB) Additional Information: full citation, abstract, references, index terms

GPSS/360 was used to model manpower allocation in the Turbine Engine Maintenance Process. The purpose was to evaluate alternate proposals for allocation of manpower. Forecasted workloads of engines were processed against these proposals. Simulation outputs included engine production times and quantities of engines produced. Also included are manpower and facility utilization outputs. Appraisal of outputs by management aided in selection of a manpower allocation plan.

20 Handling control

Dorai Sitaram
June 1993 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1993 conference on Programming language design and implementation PLDI '93**, Volume 28 Issue 6
Publisher: ACM Press
Full text available: pdf(891.97 KB) Additional Information: full citation, abstract, references, citings, index terms

Non-local control transfer and exception handling have a long tradition in higher-order programming languages such as Common Lisp, Scheme and ML. However, each language stops short of providing a full and complementary approach—control handling is provided only if the corresponding control operator is first-order. In this work, we describe handlers in a higher-order control setting. We invoke our earlier theoretical result that all denotational models of control langu ...